

mario

TODO

- prompt and validate user input
- draw the pyramid

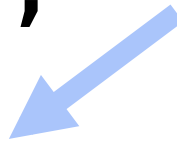
TODO

- prompt and validate user input
- draw the pyramid

do-while loops

```
int n;  
do  
{  
    n = get_int();  
}  
while (n is invalid);
```

what should this
condition be?



validating input

- valid heights: $[0, 23]$
- invalid heights: less than 0, more than 23

- design your condition carefully!

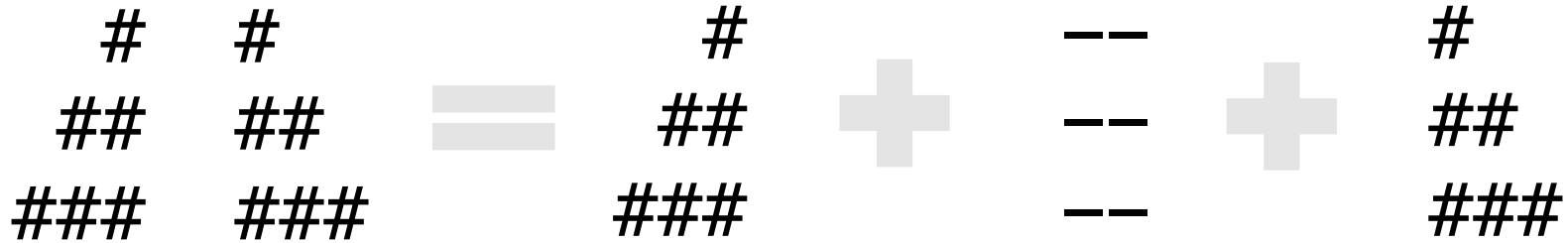
truth table

| bool1 | bool2 | bool1 && bool2 | bool1 bool2 |
|-------|-------|----------------|----------------|
| false | false | false | false |
| false | true | false | true |
| true | false | false | true |
| true | true | true | true |

TODO

- prompt and validate user input
- draw the pyramid

breaking down the full pyramid



full pyramid = left pyramid + gap + right pyramid

draw the pyramid

- for each row...
 - print left pyramid
 - print spaces
 - print hashes
 - print gap
 - 2 spaces
 - print right pyramid
 - print hashes

```
      ##  ##
     ###  ###
    ####  ####
   #####  #####
  #####  #####
 #####  #####
#####  #####
#####  #####
```

draw the pyramid

- for each row...
 - print left pyramid
 - print spaces
 - print hashes
 - print gap
 - 2 spaces
 - print right pyramid
 - print hashes

```
          #  #  
         ## ##  
        ### ###  
       #### ####  
      ##### #####  
     ##### #####  
    ##### #####  
   ##### #####  
  ##### #####  
 ##### #####
```

left pyramid pattern

height = 8

- first row: 1 #, 7 spaces
- second row: 2 #, 6 spaces
- third row: 3 #, 5 spaces
- ...
- n^{th} row: how many #?
how many spaces?
- 0-indexed or 1-indexed?

```
          #
         ##
        ###
       ####
      #####
     #####
    #####
   #####
  #####
 #####
```

gap

- always two spaces

right pyramid pattern

height = 8

- first row: 1 #
- second row: 2 #
- third row: 3 #
- ...
- n^{th} row: how many hashes?
how many spaces?

```
#  
##  
###  
####  
#####  
#####  
#####  
#####  
#####
```

draw the pyramid

- for each row...
 - print left pyramid
 - print spaces
 - print hashes
 - print gap
 - 2 spaces
 - print right pyramid
 - print hashes

```
          #  #  
         ## ##  
        ### ###  
       #### ####  
      ##### #####  
     #####  
    #####  
   #####  
  #####  
 #####
```

for loops

```
for (initialization; condition; update)  
{  
    // code inside  
}  
  
// code outside
```

for loops: repetition

initialization



```
for (int i = 0; i < 50; i++)  
{  
    printf("Hello, world!\n");  
}
```


for loops: repetition

condition

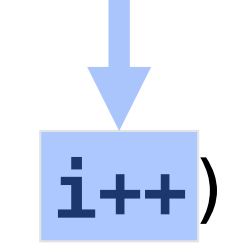


```
for (int i = 0; i < 50; i++)  
{  
    printf("Hello, world!\n");  
}
```

for loops: repetition

```
for (int i = 0; i < 50; i++)  
{  
    printf("Hello, world!\n");  
}
```

update
↓



for loops: iteration

```
for (int i = 0; i < height; i++)  
{  
    // do something for each row  
}
```

draw the pyramid

- for each row...
 - print left pyramid
 - print spaces
 - print hashes
 - print gap
 - 2 spaces
 - print right pyramid
 - print hashes

```
          #  #  
         ## ##  
        ### ###  
       #### ####  
      ##### #####  
     ##### #####  
    ##### #####  
   ##### #####  
  ##### #####  
 ##### #####
```

draw the half pyramid

```
for (int i = 0; i < height; i++)  
{  
    // print spaces for left pyramid  
    // print hashes for left pyramid  
    // print gap  
    // print hashes for right pyramid  
    // print new line  
}
```

TODO

- ☑ prompt and validate user input
- ☑ draw the pyramid

this was mario



```
      #      #  
     ##     ##  
    ###    ###  
   ####   ####  
  #####  #####  
 #####  #####  
#####  #####  
#####  #####  
#####  #####
```